Internet Engineering Task Force
Internet Draft
draft-ietf-diffserv-rsvp-00.txt

Y.Bernet, Microsoft
R. Yavatkar, Intel
P. Ford, Microsoft
F. Baker, Cisco
L. Zhang, UCLA
K. Nichols, Bay Networks
M. Speer, Sun Microsystems

June, 1998

A Framework for Use of RSVP with Diff-serv Networks

Status of this Memo

1. Abstract

In the past several years, work on QoS enabled networks led to the development of the Integrated Services (Intserv) architecture [12] and the RSVP signaling protocol [1]. RSVP, as specified, enables applications to signal per-flow requirements to the network. Intserv parameters are used to quantify these requirements for the purpose of admission control. However, as work on RSVP and Intserv has proceeded, we have recognized the following basic limitations, which impede deployment of these mechanisms in the Internet at large:

1) The reliance of RSVP on per-flow state and per-flow processing raises scalability concerns in large networks.
2) Today, only a small number of hosts generate RSVP signaling. While this number is expected to grow dramatically, many applications may never generate RSVP signaling.
3) Many applications require a form of QoS, but are unable to express these requirements using the intserv model.

At present, the market is pushing for immediate deployment of a QoS solution that addresses the needs of the Internet as well as enterprise networks. This push has led to the development of Differentiated services (diff-serv). In contrast to RSVP's per-flow orientation, diff-serv networks classify packets to one of a small number of aggregated flows, based on the setting of bits in the TOS field of each packet's IP header. Thus, in addition to eliminating the reliance on per-flow state, diff-serv QoS can initially be deployed using top-down provisioning, with no requirement for end- to-end signaling.

At the same time however, it is important to assure that the diff-serv mechanisms deployed, interoperate effectively with hosts and networks that provide per-flow QoS in response to end-to-end signaling. This is important, as we believe that in the coming years, there will be a proliferation of applications that depend on QoS and of hosts which will signal end-to-end on their behalf.

This draft proposes a framework in which diff-serv capable transit networks provide aggregate QoS services, in support of RSVP/Intserv capable hosts and stub networks, which use end-to-end signaling. In our model, diff-serv mechanisms are used within transit networks and at the boundaries between them, while either diff-serv or RSVP/Intserv mechanisms are used within stub networks and at the boundaries between stub networks and transit diff-serv networks. Managers of the transit networks will provision a pool of network resources to be available in response to end-to-end signaling. The remaining resources will be allotted using traditional 'top-down' provisioning methods.

Our framework allows the deployment of diff-serv networks and

RSVP/Intserv networks to proceed at their own pace, providing immediate incremental benefits in areas of the network in which one or the other is deployed and additional benefits where both are deployed. This framework builds upon current work in the IETF including diffserv [10] and RSVP aggregation [8].

Many of the ideas in this document have been previously discussed in the original intserv architecture document [12].

2. Goals of This Draft

This draft is based on the assumption that end-to-end QoS is required to support the needs of certain applications. Such applications include IP-telephony, video-on-demand and various non- multimedia mission-critical applications.

In our view, intserv and diff-serv are complementary tools in the pursuit of end-to-end QoS. Each serves an important purpose in the end-to-end QoS enabled network. The primary goal of this draft is to encourage the continued development of each in a manner that does not preclude realization of the proposed framework. To this end, we will:

1. List the requirements of a network that provides end-to-end QoS.
2. Present a framework that uses intserv as a customer of diff-serv to meet these requirements.
3. Identify dependencies of intserv on diff-serv.

Ultimately, we aim to clearly define a manner in which RSVP/Intserv and diff-serv mechanisms interact seamlessly. We expect that by doing so, we will enable network administrators to determine the degree to which diff-serv capabilities are pushed towards the edge of their networks (or, the degree to which RSVP/Intserv capabilities are pushed towards the core).

3. Terminology

The following terms are used in this draft:

a. Intserv region (or intserv capable network) - the part of an internet that uses per-flow identification, signaling, and admission control to deliver per-flow QoS guarantees

b. Diff-serv region (or diff-serv capable network) - the part of an internet that provides aggregate QoS services

c. Quantitative QoS applications - applications for which QoS

requirements are readily quantifiable, and which rely on these QoS requirements to function properly.

d. Qualitative QoS applications - applications for which relative QoS requirements exist, but are not readily quantifiable.

e. QoS applications - applications that require some form of QoS, either qualitative or quantitative.

f. Loose QoS - QoS assurances which are relative, rather than absolute, or generally not quantifiable.

g. Tight QoS - QoS assurances which are absolute and quantifiable, though they may or may not provide 100% guarantee.

h. Top-down (or open-loop) provisioning - traditional provisioning methods which configure network capacities using heuristics and experience, typically from a console, with no explicit knowledge of exact traffic volumes or exact paths taken by the affected traffic.

i. RSVP/Intserv - RSVP is a signaling protocol. Intserv (in this context) is a model for quantifying traffic that is useful for admission control purposes. In this document, we use the terms together, to discuss the RSVP/Intserv network, in contrast to the diff-serv network. However, the two are separable and much of the following discussion could be applied to a model in which RSVP signals using parameters that are not Intserv specific.

4. Requirements for the End-to-End QoS Framework An end-to-end QoS network must serve the requirements of network managers as well as those of both quantitative and qualitative QoS applications. We consider these requirements in the context of the following general topology:

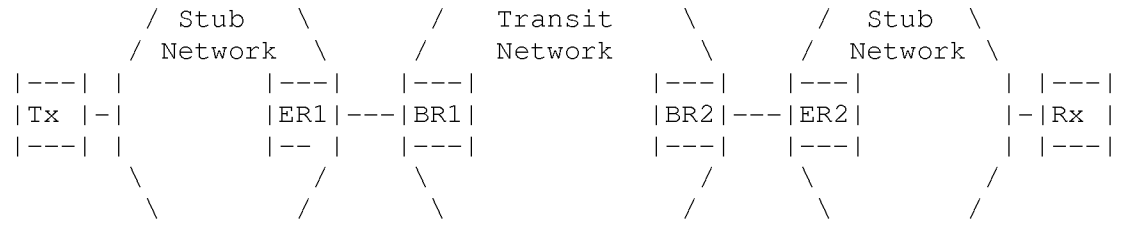

Figure 1: Sample Network Configuration

This network consists of a diff-serv capable transit network and two

intserv capable stub networks. In the interest of simplicity, we show a single QoS sender on one of the stub networks and a single QoS receiver on the other. We show edge routers (ER1, ER2) at the interfaces of the intserv networks to the diff-serv network. We show boundary routers (BR1, BR2) at the interfaces of the diff-serv network to the intserv networks.

The transit network contains a mesh of routers, at least some of which are diff-serv capable. The stub networks contain a mesh of routers, at least some of which are intserv capable.

We define the following requirements of the framework:

4.1 Definition of a Set of Services

There must be a set of useful end-to-end services available to Quantitative QoS applications. Routers internal to the diff-serv network are assumed to provide a set of 'per-hop-behaviours' (PHBs [10]). We expect that concatenation of certain well-defined PHBs will yield certain well-understood services across the diff-serv network. We also expect that the intserv regions of the network will be able to extend these services such that they can be realized in a true end-to-end manner.

In addition, there must be a set of end-to-end services available to Qualitative QoS applications. However, the services that these applications require are generally less demanding. They can be loosely provisioned (in a top-down manner) in the diff-serv regions of the network and will likely receive best-effort treatment in the intserv regions of the network.

In this draft we focus primarily on the requirements of quantitative QoS applications. Although these may generate only a small fraction of all traffic, servicing this traffic may comprise a significant fraction of the revenues associated with QoS. In addition, while qualitative QoS applications can be satisfied by conventional diff- serv alone, quantitative QoS applications require additional support.

4.2 Allotment of Diff-serv Service Levels to Specific Traffic Flows

It must be possible for QoS applications to invoke specific end-to-end service levels for their traffic flows. Within the intserv regions of the network quantitative QoS applications do so by using RSVP signaling to configure classifiers which operate on IP addresses and port numbers. We will refer to such classifiers from here on as 'MF' classifiers [10].

Within the diff-serv regions of the network, traffic is allotted

service based on the contents of the DS-field in packet headers. Therefore, it is necessary for QoS applications to effect the correct marking of DS-fields before their packets are submitted to the diff-serv network. (This is particularly true for quantitative QoS applications, less so for qualitative QoS applications that need not play as active a role in securing specific QoS from the network). There are two general mechanisms for doing so:

1. Hosts may directly mark DS-fields in the transmitted packets of QoS applications.
2. Routers external to the diff-serv network may mark DS-fields on behalf of QoS applications based on MF classification.

In the first case, marking will be done based on host configuration or local communication between QoS applications and the host operating system. In the second case, marking will be done based on top-down configuration of the marking router's MF classifier/marker (by manual configuration or via automated configuration scripts) or based on standard signaling between QoS applications and the marking router's classifier/marker.

The following three requirements argue either for host based marking or for dynamic configuration of a router's classifier/marker in response to application requests.

4.2.1 Minimal Management Burden

The information required to express useful mappings of application traffic flows to service levels is likely to be quite complex and to change frequently. Thus, manual configuration is likely to impose a significant management burden. If the configuration information is very simple and does not change over time, the management burden may be relatively minor. However, this means that the granularity of allotting service levels to flows will be sub-optimal.

4.2.2 Granularity of Allotment

The term 'granularity' is used here to refer to the degree of specificity that is available in allotting a specific service level to a specific traffic flow. There are two measures of granularity; one is the granularity with which an individual flow or a group of flows is identified. The other is the frequency at which the service allotted to a flow may change. A fine grain QoS system would allow the following requirement to be expressed: telephony traffic from user X should be allotted service level A, while telephony traffic from user Y should be allotted service level B, and web traffic from any user should be allotted service level C. A coarse grain system would be limited to something of the form: all traffic from subnet 1.0.0.0

should receive service level A, while all traffic from subnet 2.0.0.0 should receive service level B. A temporally fine grain system would allow immediate changes in allotment of service levels to traffic flows. A temporally coarse grain system may allow infrequent changes only.

4.2.3 Difficulties in Identification of Application Flows

Routers may not be able to readily identify specific application flows based on network and/or transport layer fields in a packet.

For example, consider the need to give preferential service to a website's home page (over other, less important pages at the site) or the case of encrypted traffic flows (IPSEC).

4.3 Admission Control

Quantitative QoS applications use RSVP to request that their flows be admitted to intserv regions of the network. When a request is rejected, the host application may avoid sending traffic and/or intermediate RSVP capable nodes will only give best-effort service to traffic on flows that were not admitted. These mechanisms protect traffic on flows that were admitted.

In diff-serv regions of the network, admission control is provided implicitly, by policing at ingress points based on provisioning. The problem with implicit admission control is that it breaks the end-to-end validity of explicit admission control. Specifically, an application may gain admission using RSVP signaling, even though there is no capacity for that application's traffic within the diff-serv region of the network. Neither the application, nor intermediate RSVP capable nodes will be aware that the application's traffic is not admissible. As a result, neither can take corrective action and all traffic from that customer, at the corresponding service level, may be compromised. This failure may be partially, but not completely alleviated by policing based on MF classification at the diff-serv ingress (rather than BA classification [10]).

End-to-end QoS requires that quantitative QoS applications and RSVP capable intserv nodes be explicitly informed of admission control failure in the diff-serv network. This enables them to take corrective action and to avoid overdriving the diff-serv network. If the service agreement between the intserv and diff-serv regions of the network is statically provisioned, then admission control functionality can be provided by static configuration of admission control in intserv edge routers. However, if the service agreement is dynamically variable, then it will be necessary to dynamically propagate current diff-serv resource availability to the intserv network for the purpose of

explicit admission control.

4.4 Policy Support

End-to-end QoS leads to preferential treatment of certain traffic flows over others. Within diff-serv regions of the network, policy applies on a per-customer basis. In general, the diff-serv network makes multiple service levels available to a single customer's intserv network. In this case the customer must apply policy within its network to assure appropriate allocation of resources (customer network resources as well as diff-serv network resources) to individual hosts in the customer's network. This requires that end- to-end admission control be based on policy as well as resource availability.

5. Intserv as a Customer of Diff-serv

To meet the above requirements, we envision a network that consists typically of relatively smaller, intserv capable stub networks, connected by larger, diff-serv capable transit networks. In this section, we will describe the operation of one instantiation of such a network (see figure 1). The following assumptions apply:

5.0.1 Host Capabilities

Both sending and receiving hosts use RSVP to communicate QoS requirements of certain QoS aware applications running on the host. A QoS process within the host operating system generates RSVP signaling on behalf of the applications. This process also invokes traffic control. Host traffic control includes marking the DS-field in transmitted packets and shaping transmitted traffic per token bucket specifications. Note that host traffic control is assumed for this specific example, but is not a requirement of the framework in general. Leaf routers within the intserv network may provide the traffic control functions.

5.0.2 Edge Routers

The edge routers are special routers that straddle the boundary between the RSVP/Intserv region of the network and the diff-serv region of the network. It is helpful to think of these routers as consisting of two halves; the standard RSVP half, which interfaces to the stub networks, and the diff-serv half, which interfaces to the transit network.

The RSVP half is at least partially RSVP capable; it is able to process PATH and RESV messages but it is not necessarily required to store full RSVP state and it is not required to provide MF classification.

The diff-serv half of the router provides the interface to the admission control function for the diff-serv network. We shall refer to this function from here on as the 'DACS' (diff-serv admission control service). The DACS is a process that is likely to (but is not required to) run at least partially, on the routers. If the service agreement between the stub networks and the transit networks is statically provisioned then the DACS can be as simple as a table which specifies capacity at each service level. If the service agreement is dynamic, the DACS may communicate with counterparts within the diff-serv network (such as a bandwidth broker [4]) and may be able to make admission control decisions based on provisioned limits as well as the topology and the capacity of the diff-serv network.

5.0.3 Boundary Routers

These are conventional boundary routers. In the example illustrated, they are not required to run RSVP. They are expected to implement the policing function of diff-serv ingress routers, based on the results of a BA classifier. They may, but are not required, to provide MF classification nor to mark the DS-field (with the possible exception of the in/out bit). [10, 8]

Note that this example places the boundary between the RSVP/Intserv network and the diff-serv network, within the edge routers at the stub networks. In general, this boundary could be shifted to the left or to the right. It could for example, be placed within the boundary routers in the transit network. In this case, the DACS is implemented entirely within the diff-serv network (and is essentially, the bandwidth broker proposed in [4]), but the diff- serv boundary routers must be RSVP capable.

5.0.4 Stub Networks

The stub networks consist of int-serv capable hosts and some number of leaf routers. Leaf routers within the stub networks may or may not be int-serv capable. Since they are relatively small networks, it is reasonable to assume that they are int-serv capable, but this is not necessary. If they are not int-serv capable, we assume that they are not capable of per-flow identification, signaling, and admission control and, in that case, will pass RSVP messages (requesting per-flow QoS) unhindered.

5.0.5 Transit Network

The transit network is not capable of per-flow identification, signaling, and admission control. It provides two or more levels of service based on the DS-field in the headers of carried packets (diff-serv capable). Furthermore, the transit network is able to carry RSVP

messages transparently, with minimal or no impact on its performance (see [8]). The transit network may include multiple carrier networks.

5.0.6 Carrier/Customer Agreement

The customer (owner(s) of the leaf networks) and the carrier owning the transit network have negotiated a contract for the capacity to be provided at each of a number of standard diff-serv service levels. The capacity may be statically provisioned. In this case, the DACSs are statically configured with the capacity available at each service level and reside entirely within the edge routers. Alternatively, the capacity may be dynamically variable with a pre- negotiated usage fee and/or a pre-negotiated capacity limit. In this case, the DACS would be required to communicate with counterparts within the diff-serv transit network.

5.0.7 Mapping from Intserv Service Type to DS-field

In our proposal, we use RSVP signaling to provide admission control to specific service levels in the diff-serv, as well as the intserv network. RSVP signaling requests carry an intserv service type, describing the type of service they expect from the intserv regions of the network. At each hop in an intserv network, the generic intserv service requests are interpreted in a form meaningful to the specific media.

For example, at an ATM hop, a VC of the correct type (CBR, ABR or VBR) is established [13]. At an 802.1 hop, the intserv service type is mapped to an appropriate 802.1p priority level [5]. At the boundary between the intserv network and a diff-serv network, it is necessary for edge devices to map the requested intserv service to a diff-serv service level that can reasonably extend the intserv service type requested by the application. The edge device can then provide admission control to the diff-serv network by accepting or rejecting the request based on the capacity available at the requested diff-serv service level.

We assume that one of two schemes is used to map intserv service types to diff-serv service levels. In the first scheme (called "default mapping"), we propose a standard, well-known mapping from intserv service type to a PHB that will invoke the appropriate behavior in the diff-serv network. The mapping is not necessarily one-to-one. For example, controlled-load interactive voice traffic will likely map to a PHB having different latency characteristics than controlled-load latency tolerant traffic. For this reason we suggest adding a qualifier to the intserv service type indicating its relative latency tolerance (high or low). The qualifier would be defined as a standard object in intserv signaling messages.

In an alternate scheme (called "customer-specified mapping"), we allow

the devices at the edge of the diff-serv region of the network to modify the well-known mapping. Under this approach, RESV messages originating at hosts carry the usual intserv service type (with a qualifier, as described above). When RESV messages arrive at the interface of the int-serv and diff-serv regions (e.g. router ER1 in Figure 1, where the traffic from the stub network enters the diff-serv region), the edge device will determine the PHB that should be used to obtain the corresponding diff-serv service level. This value is appended to the RESV message by the edge device and is carried to the sending host. When the RESV message arrives at the sending host, the sender (or intermediate intserv routers) will mark outgoing packets with the indicated PHBs.

The decision to modify the well-known mapping at the edge devices will be based on edge-device configuration and/or policy decision at the edges.

5.1 How End-to-End QoS is Obtained

The following sequence illustrates the process by which an application obtains end-to-end QoS:

1. The sending host's QoS process generates an RSVP PATH message, describing the traffic offered by the sending application.

2. The PATH message is carried toward the receiving host. In the sending stub network, standard RSVP processing will be applied at RSVP capable nodes (routers, SBMs, etc).

3. At ER1, the PATH message is subjected to standard RSVP processing and PATH state is installed in the router. The PATH message is sent onward, to the transit network.

4. The PATH message is carried transparently through the transit network. It is processed in the receiving stub network according to standard RSVP processing rules.

5. At the receiving host, the QoS process generates an RSVP RESV message, indicating interest in the offered traffic, at a certain intserv service level.

6. The RESV message is carried back towards the sending host. Consistent with standard RSVP processing, it may be rejected at any RSVP node in the receiving stub network if resources are deemed insufficient to carry the traffic requested.

7. At ER2, the RESV message is subjected to standard RSVP

processing. It may be rejected if resources on the downstream interface of ER2 are deemed insufficient to carry the resources

requested. If it is not rejected, it will be carried transparently through the transit network, arriving at ER1.

8. At this point, the RESV message triggers DACS processing. The DACS compares the resources requested to the resources available at the corresponding diff-serv service level, in the diff-serv enabled transit network. If the RESV message is admitted, the DACS updates the available capacity for the service class, by subtracting the approved resources from the available capacity.

9. Assuming the available capacity is sufficient, the RESV message is admitted and is allowed to continue upstream towards the sending host. If the available capacity is insufficient, the RESV message will be rejected and the available capacity for the service class will remain unchanged.

10. The RESV message proceeds through the sending stub network. RSVP nodes in the sending stub network may reject it. If it is not rejected, it will arrive at the sending host.

11. At the sending host, the QoS process receives the RESV message. It interprets receipt of the message as an indication that the specified traffic has been admitted for the specified intserv service type (in the RSVP enabled regions of the network) and for the corresponding diff-serv service level (in the diff-serv enabled regions of the network). It begins to set the DS-field in the headers of transmitted packets, to the value which maps to the Intserv service type specified in the admitted RESV message.

In this manner, we are able to obtain end to end QoS through a combination of networks that support RSVP style reservations and networks that support diff-serv style priortization. The successful arrival of RESV messages at the original sender indicates that admission control has succeeded both in the RSVP regions of the network and in the diff-serv regions of the network.

5.2 Variations of the Model

It is useful to consider a number of variations of the model presented.

5.2.1 Admission Control

5.2.1.1 Statically Provisioned Service Agreements

In the simplest model, service agreements are negotiated statically between the stub networks and the transit networks. A service agreement consists of a table of capacities available to a customer's stub

network, at each diff-serv service level. In this case, DACS functionality is provided at the edge routers in the stub networks. The 'diff-serv half' of these routers appear to the 'RSVP half' as a sending interface with resource limits defined by the service agreement table. While there may be bandwidth brokers and dynamic provisioning within the transit networks, these are not coupled with the intserv stub networks and admission control in the two regions of the network is completely independent.

5.2.1.2 Dynamic Service Agreements

In a more sophisticated model, service agreements between customer stub networks and carrier transit networks are more dynamic. Customers may be able to dynamically request changes to the service agreement. In this case, a statically provisioned edge router cannot provide the required DACS functionality. Instead, DACS functionality must be provided by coupling the stub network's admission control with the transit network's admission control.

The two admission control mechanisms meet at the boundary between the diff-serv network and the intserv network. This boundary may be implemented at the edge router (in the stub network), at the boundary router (in the transit network), or at the bandwidth broker for the intserv network.

5.2.1.3 Limiting the Impact of Intserv Admission Control on the Diff-serv Network

Note that coupling intserv and diff-serv admission control does not imply that each intserv admission control request results in diff-serv admission control work. Instead, intserv admission control requests are aggregated at the boundary between the intserv and the diff-serv network. For example, intserv admission control requests may trigger diff-serv admission control requests to bandwidth brokers only when some high-water or low-water resource threshold is crossed. Separate high-water and low-water thresholds provide hysteresis to prevent thrashing.

5.2.1.4 Roles of Policy and Resource Based Admission Control

It is necessary to provide both resource and policy based admission control in the diff-serv network as well as the intserv network. In the diff-serv network, resource and policy based admission control are handled by entities such as bandwidth brokers and reflected to the intserv network as DACS (or RSVP). Policy decisions made within the

diff-serv network are likely to be at the per-intserv network (per-customer of the diff-serv network) granularity.

In the intserv network, resource based admission control is handled by

RSVP enabled routers (and SBMs [2]). Policy based admission control is handled by RSVP capable policy servers. These assure that intserv resources are allotted to intserv customers according to policy specific to the intserv network. In addition, policy servers within the intserv network must also assure that appropriate policy is applied when diff-serv resources are allotted to intserv customers.

5.2.2 Setting the DS-field at Intermediate Nodes

In the example described, hosts use RSVP signaling and mark the DS-byte corresponding to the admitted service level. Note that these functions can be separated. In the example, the function of RSVP signaling is to invoke QoS in the intserv network and to provide end-to-end admission control. The function of marking the DS-field is to reduce the need for MF classification at routers. (MF classification is required at the ingress to a diff-serv network only to determine the customer to whom the traffic belongs. If an interface is dedicated to the customer, no MF classification need be done. In this case, any MF classification on behalf of the diff-serv ingress point is provided as a service to the customer and goes beyond policing requirements).

It is possible to mark the DS-field at intermediate routers rather than at the host and still to realize many of the benefits of our approach. In this case, intermediate routers may use the RSVP signaling to configure an MF classifier and marker. Therefore, the configuration of MF classifiers and markers is dynamic (minimizing the management burden) and full resource and policy based admission control can be applied.

The disadvantages of marking the DS-field at intermediate routers (instead of the host) are that full MF classifiers are required at the intermediate nodes and that responsibility for traffic separation is shifted away from the host.

Nonetheless, this approach is necessary to support those hosts which may be capable of RSVP signaling, but which are not capable of marking the DS-field. In addition, there may be cases in which the network administrators wish to shift the responsibility for traffic separation away from the hosts. In particular, we expect that there will continue to be a need for top-down provisioned MF classification, especially for qualitative (as opposed to quantitative) QoS applications.

6. Managing Different Resource Pools

We have focused largely on the class of applications that use RSVP to explicitly signal per-flow QoS requirements and which expect end- to-end tight QoS assurances, spanning both the intserv and diff-serv regions of the network. We have been referring to these applications as 'quantitative QoS applications'.

However, diff-serv networks also provide loose QoS to applications that do not explicitly signal. Network managers can allot qualitative QoS applications specific QoS in the diff-serv network. This is achieved by configuring classifiers at the ingress to the diff-serv network to recognize traffic from these applications. Thus, the classification fields are used as a form of implicit signaling.

Network administrators must therefore share diff-serv network resources between three types of traffic:

a. Quantitative (explicitly signaled) QoS application traffic
b. Qualitative (implicitly signaled) QoS application traffic
c. All other (best-effort) traffic

Quantitative QoS applications rely on explicit admission control for their traffic, at the edges of the diff-serv network. This traffic may be refused admission for a particular diff-serv service level. However - if admitted, the traffic is assured tight QoS. Of course, this is true only to the extent that, at any ingress point, the total offered traffic at each service level does not exceed the resources requested through the sum of admission control requests.

Traffic from qualitative QoS applications is provided with implicit admission control as a result of policing at ingress points. However, implicit admission control does not provide explicit feedback to applications. Therefore, it is difficult to assure that the total traffic offered at an ingress point will not exceed the levels allowed by policers. Thus, traffic from qualitative applications is offered only loose QoS.

From the network manager's perspective, there are three pools of resources in the diff-serv network; one for traffic sourced by quantitative QoS applications, one for traffic sourced by qualitative QoS applications and one for best-effort traffic. These pools must be isolated from each other by the appropriate configuration of policers and classifiers at ingress points to the diff-serv network, and by appropriate provisioning within the diff- serv network.

7. Issues

7.1 Setting the DS-field at Hosts

The thought of allowing hosts to set the DS-field directly, may alarm network administrators. The obvious concern is that hosts may attempt to 'steal' resources. In fact, hosts may attempt to exceed the negotiated capacity at a particular service level regardless of whether they invoke this service level directly (by setting the DS- byte) or indirectly (by submitting traffic that classifies in an intermediate

router to a particular diff-serv PHB).

In either case, it may be necessary to protect the network by policing at various points, both within the stub network and/or at the interface to the transit network. For example, within the stub network, routers may police the aggregate traffic coming from a host to ensure that the host is not exceeding its traffic limit. This assures protection against malicious users or malfunctioning equipment and, overall, ensures that customers do not use more resources than they are entitled to, at each service level. If the sending host does not do the marking, intermediate and/or boundary routers must provide MF classification, mark and police. If the sending host does do the marking, these routers need only to provide BA classification and to police the aggregate to ensure that the customer is not exceeding the aggregate capacity negotiated for the service level.

Requiring hosts to mark the DS-field has the effect of moving responsibility to the edge of the network, in more ways than one. With this approach, boundary routers police in aggregate. As a result, the customer cannot rely on boundary routers to provide traffic isolation between the customer's flows, when policing or shaping. Instead, it is the customer's responsibility to ensure that the customer's flows are properly shaped and policed within the customer's sending network. Overall, this approach simplifies boundary routers and still allows protection against misbehaving hosts/users.

Ideally, hosts should provide per-flow shaping at their sending interfaces. However, there is always a chance that the customer's traffic will become distorted as it nears the boundary between the customer and the carrier. In this case, the customer should do per flow policing (or even re-shaping) at the egress point from the customer's network unless the policing agreement at the other side is known to accommodate the transient bursts that can arise from adding the flows.

In summary, the security concerns of marking the DS-field at the edge of the network can be dismissed since each carrier will have to do some for of policing (per-flow or per-host) at their boundary anyway. Furthermore, this approach reduces the granularity at which boundary routers must police, thereby pushing finer grain shaping and policing responsibility to the edges of the network, where it scales better. The carriers are thus focused on the task of protecting their transit

networks, while the customers are focused on the task of shaping and policing their own traffic to be in compliance with their negotiated token bucket parameters.

7.2 End-to-End Integrity of the DS-field

Our proposal assumes that hosts use a standard coding for specifying a

desired PHB in some sub-field of the DS-field. It also assumes that packets submitted to the network with a certain PHB specified, will receive a standard service throughout the diff-serv network. Strictly speaking, this does not dictate that the transit network must leave the PHB field intact. However, we see little value in allowing the PHB field to be altered within the network. This is likely to perpetuate local and incompatible interpretations of the field.

7.3 Carrying RSVP Messages across Transit Networks

Our proposal presumes end-to-end RSVP both as a means for communication between sending host and receiving host and optionally, for the support of true RSVP reservations in stub networks (or in intermediate networks which are interested in the fine grain RSVP information). Therefore, we require that RSVP messages be carried transparently across the transit networks. In [8] mechanisms are proposed for doing so in a manner that does not require the routers in the transit network to understand/interpret RSVP messages and does not adversely impact the transit network.

8. Dependencies of Intserv on Diff-Serv

We have described a framework for end-to-end QoS in which intserv networks are customers of diff-serv networks. We believe that the benefits of this framework are sufficient to justify the consideration of intserv dependencies as diff-serv work proceeds. In particular, we wish to draw attention to the following dependencies:

1. We expect that we can invoke a standard end-to-end (within the diff-serv network) service by specifying a standard code in a (PHB) sub-field of the DS-field of a packet launched into a diff-serv network.

2. Diff-serv networks must provide admission control information to the intserv network. At the very least, this is through static service level agreements. Preferably, this is through a dynamic protocol. If the intserv to diff-serv boundary is implemented in the transit network boundary routers, then this protocol is RSVP.

3. We expect that diff-serv networks will transparently carry RSVP messages such that they can be recovered at the egress point from the diff-serv

network.

9. Security Considerations

We are proposing that RSVP signaling be used to obtain resources in both the diff-serv and intserv regions of the network. Therefore, all RSVP security considerations apply [11]. In addition, network adminis-

trators are expected to protect network resources by configuring secure policers at interfaces with untrusted customers.

10. References

[1] Braden, R., Zhang, L., Berson, S., Herzog, S. and Jamin, S., "Resource Reservation Protocol (RSVP) Version 1 Functional Specification", RFC 2205, Proposed Standard, September 1997

[2] Yavatkar, R., Hoffman, D., Bernet, Y., Baker, F. and Speer, M., "SBM (Subnet Bandwidth Manager): A Protocol For RSVP-based Admission Control Over IEEE 802 Style Networks", Internet Draft, March 1998

[3] Berson, S. and Vincent, R., "Aggregation of Internet Integrated Services State", Internet Draft, December 1997.

[4] Nichols, K., Jacobson, V. and Zhang, L., "A Two-bit Differentiated Services Architecture for the Internet", Internet Draft, December 1997.

[5] Seaman, M., Smith, A. and Crawley, E., "Integrated Services Over IEEE 802.1D/802.1p Networks", Internet Draft, June 1997

[6] Elleson, E. and Blake, S., "A Proposal for the Format and Semantics of the TOS Byte and Traffic Class Byte in Ipv4 and Ipv6 Headers", Internet Draft, November 1997

[7] Ferguson, P., "Simple Differential Services: IP TOS and Precedence, Delay Indication, and Drop Preference", Internet Draft, November 1997

[8] Guerin, R., Blake, S. and Herzog, S., "Aggregating RSVP based QoS Requests", Internet Draft, November 1997

[9] Clark, D. and Wroclawski, J., "An Approach to Service Allocation in the Internet", Internet Draft, July 1997

[10] Blake, S. and Nichols, K., "Differentiated Services Operational Model and Definitions", Internet Draft, February 1998

[11] Baker, F., "RSVP Cryptographic Authentication", Internet Draft, August 1997

[12] Braden, R., Clark, D. and Shenker, S., "Integrated Services in the Internet Architecture: an Overview", Internet RFC 1633, June 1994

[13] Garrett, M. W., and Borden, M., "Interoperation of Controlled-Load Service and Guaranteed Service with ATM", Internet Draft, March

1998

11. Acknowledgments

12. Author's Addresses

Yoram Bernet
Microsoft
One Microsoft Way,
Redmond, WA 98052
Phone: (425) 936-9568
Email: yoramb@microsoft.com

Raj Yavatkar
Intel Corporation, JF3-206
2111 NE 25th. Avenue,
Hillsboro, OR 97124
Phone: (503) 264-9077
Email: yavatkar@ibeam.intel.com

Peter Ford
Microsoft
One Microsoft Way,
Redmond, WA 98052
Phone: (425) 703-2032
Email: peterf@microsoft.com

Fred Baker
Cisco Systems
519 Lado Drive,
Santa Barbara, CA 93111
Phone: (408) 526-4257
Email: fred@cisco.com

Lixia Zhang
UCLA
4531G Boelter Hall
Los Angeles, CA 90095

Phone: +1 310-825-2695
Email: lixia@cs.ucla.edu

Kathleen Nichols
Bay Networks
Email: Kathleen_Nichols@BayNetworks.COM

Michael Speer
Sun Microsystems, Inc
901 San Antonio Road UMPK15-215

Palo Alto, CA 94303
phone: +1 650-786-6368
Email: speer@Eng.Sun.COM